ICS 311: Algorithms Spring 2025

Problem Set 12

Prof. Nodari Sitchinava *Due: Friday, Apr 11, 2025 at 4pm*

You may discuss the problems with your classmates, however **you must write up the solutions on your own** and **list the names** of every person with whom you discussed each problem.

Start **every** problem on a separate sheet of paper, with the exception of Problems 1 (Peer credit assignment) – Problem 1 can be on the same page as any other problem. Any problem that starts on the same sheet of paper as some other problem will receive 0 points!

1 Peer Credit Assignment (1 point extra credit for replying)

Please list the names of the other members of your peer group for this week and the number of extra credit points you think they deserve for their participation in group work.

- You have a total of 60 points to allocate across all of your peers.
- You can distribute the points equally, give them all to one person, or do something in between.
- You need not allocate all the points available to you.
- ***You cannot allocate any points to yourself!*** Points allocated to yourself will not be recorded.

2 Beach Meetup (30 pts)

Two friends, each chillin at different beaches, post pictures of their day on social media. After seeing each other's posts, they decide to meet up to enjoy the beach together. Both of them are ready to leave and start driving immediately, however, they are unable to decide on a beach to meet up at. In order to maximize the amount of time they have to hangout together, they decide to choose a beach that allows them to meet each other *as soon as possible*. Assuming they both follow their GPS properly and take the shortest route possible to the beach, which beach should they meet at?

The island is represented as a connected directed graph $G = (V, E)$, where each vertex is a beach and each directed edge represents a route from one beach to another. The time it takes to travel along each route is represented by a weight function $w : E \to \mathbb{Z}^+$, where $w(e)$ gives the travel time (in minutes) for the route e.

Given the two starting beaches, represented by vertices s_1 and s_2, where each friend is initally located, design an algorithm BEACHMEETUP(G, w, s_1, s_2) that determines the best beach for them to meet at. (If there are multiple best beaches, then return any one of them.) Write down the pseudocode, argue why your algorithm returns the correct answer, and analyze the runtime of your solution. ***All three parts must be present to receive any credit.***

3 Johnson's Algorithm (30 pts)

Suppose graph G contains a cycle C whose weight is 0. Prove that Johnson's algorithm sets $\hat{w}(u, w) = 0$ for every edge (u, w) in C.

4 Transitive Closure on Directed Graphs (40 pts)

(a) **(20 pts)** Give an $O(V^2 + VE)$-time algorithm for computing the transitive closure of a directed graph $G = (V, E)$. Write down the pseudocode, argue why it is correct and analyze its running time. ***All three parts must be present to receive any credit.***

(b) **(20 pts)** Prove that any transitive closure algorithm on a directed graph $G = (V, E)$ will take at least $\Omega(V^2)$ time even when $E = O(V)$. (Hint: provide an instance of a graph on which any algorithm will run in $\Omega(V^2)$ time.)

5 Verifying Shortest Paths (OPTIONAL - 0 pts)

You are organizing a programming competition, where contestants implement Dijkstra's algorithm. Given a directed graph $G = (V, E)$ with integer-weight edges and a starting vertex $s \in V$, their programs are supposed to output triplets $(v, v.d, v.\pi)$ for each vertex $v \in V$. Design an $O(V + E)$ time algorithm that takes as input the original graph G in ***both*** adjacency matrix $(G.M)$ and adjacency list $(G.Adj)$ representations, starting vertex s, and the output of a contestant's program (given as an array A of triplets), and returns whether A is the correct output for G. Write down the ***pseudocode*** for your algorithm, ***explain*** why it correctly verifies the output, and ***analyze*** your algorithm's running time. ***All three parts must be present to receive any credit.*** You may assume that all edge weights of the input graph provided to the contestants are nonnegative and A (the output of their programs) is in the valid format, i.e., you don't need to verify that A is actually an array of triplets, with v and $v.\pi$ being valid vertices and $v.d$ being an integer.

Can you achieve $O(V + E)$ time if the input graph is given only in the adjacency matrix or only in the adjacency list representation?

6 Dijkstra with Negative Weight Edges (OPTIONAL - 0 pts)

Suppose that we are given a weighted, directed graph $G = (V, E)$ in which edges that leave the source vertex s may have negative weights, all other edge weights are nonnegative, and there are no negative-weight cycles. Argue that Dijkstra's algorithm correctly finds shortest paths from s in this graph.

7 Detecting Negative Edge Cycles (OPTIONAL - 0 pts)

Let $G = (V, E)$ be a weighted, directed graph with a source vertex $s \in V$ that has been initialized by INITIALIZE-SINGLE-SOURCE(G, s). Prove that if a sequence of relaxation steps sets $s.\pi$ to a non-nil value, then G contains a negative-weight cycle. Can you rely on this method to detect negative-weight cycles in a graph?